

Virtual Study Buddy Platform (PeerLynx)

Project Report

Project Work Phase - II (ECS899)

Degree

BACHELOR OF TECHNOLOGY - CSE ADCA (IBM)

PROJECT GUIDE:

Mr. Amit Singh
Mr. Alok Sharma

SUBMITTED BY:

Ilma Naaz (TCA2157017)
Mohd Asif (TCA2157025)
Rohan Ray (TCA2157031)

June, 2025



Accredited with NAAC **A** Grade

12-B Status from UGC

FACULTY OF ENGINEERING & COMPUTING SCIENCES
TEERTHANKER MAHAVEER UNIVERSITY, MORADABAD

DECLARATION

We hereby declare that this Project Report titled Virtual Study Buddy Platform (PeerLynx) submitted by us and approved by our project guide, College of Computing Sciences and Information Technology. Teerthanker Mahaveer University, Moradabad, is a bonafide work undertaken by us and it is not submitted to any other University or Institution for the award of any degree diploma/certificate or published any time before.

Project ID : Project Id

Student Name: Ilma Naaz Signature

Student Name: Rohan Ray Signature

Student Name: Mohd Asif Signature

Project Guide : Mr. Amit Singh
Mr. Alok Sharma Signature

Contents

1	PROJECT TITLE	4
2	PROBLEM STATEMENT	4
3	PROJECT DESCRIPTION	4
3.1	SCOPE OF THE WORK	4
3.2	PROJECT MODULES	6
3.3	CONTEXT DIAGRAM (HIGH LEVEL)	9
4	IMPLEMENTATION METHODOLOGY	11
5	TECHNOLOGIES TO BE USED	17
5.1	SOFTWARE PLATFORM	17
5.2	HARDWARE PLATFORM	17
5.3	DEVELOPMENT TOOLS	17
6	ADVANTAGES OF THIS PROJECT	18
7	ASSUMPTIONS	21
8	FUTURE SCOPE AND FURTHER ENHANCEMENT OF THE PROJECT	23
9	PROJECT REPOSITORY LOCATION	26
10	DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	26
11	CONCLUSION	27

_Toc199763364

Appendix

A: Data Flow Diagram (DFD)	29
B: Entity Relationship Diagram (ERD)	32
C: Use Case Diagram (UCD)	33
D: Data Dictionary (DD)	34
E: Screen Shots	35

1 Project Title

Virtual Study Buddy Platform (PeerLynx)

2 Problem Statement

In the current era of online learning and remote collaboration, students frequently encounter difficulties in locating dependable study partners, coordinating group discussions, and maintaining consistent and meaningful communication. The limitations of traditional educational platforms—such as the absence of real-time interaction, lack of structured discussion spaces, and minimal personalization—have contributed to decreased motivation, reduced engagement, and a fragmented learning experience.

These challenges are particularly significant in collaborative learning environments, where effective peer interaction plays a vital role in academic success. Without a centralized, interactive platform that supports seamless collaboration, students often resort to using disjointed communication tools, leading to inefficiencies and a lack of cohesion in group efforts.

To overcome these barriers, this project proposes the development of a Virtual Study Buddy Platform that reimagines how students connect, communicate, and collaborate online. Drawing inspiration from modern communication platforms, this solution aims to provide a unified, engaging, and student-focused virtual space designed to foster community-driven learning. By enhancing the digital learning environment, the platform seeks to improve academic collaboration, peer engagement, and overall learning outcomes in a virtual setting.

3 Project Description

3.1 Scope of the Work

The Virtual Study Buddy Platform is designed to address the increasing need for collaborative and interactive digital learning spaces for students. This project provides a comprehensive solution that not only facilitates communication among students but also enhances productivity and peer learning in an online environment.

The scope of this project spans the complete development lifecycle of a collaborative study platform, including user interaction design, real-time communication features, and secure group collaboration tools.

Below are the key components of the project scope:

User Registration and Authentication

- Implement a secure user registration system with email verification.
- Allow users to log in using credentials and maintain authenticated sessions using JWT (JSON Web Tokens).
- Include options for password reset and profile protection through encryption.

Profile Management

- Each user can manage a personal profile including basic details (name, email), academic interests, and availability.
- Enable profile customization to help students find suitable study buddies or group matches.

Study Group Creation and Management

- Users can create new study groups or join existing ones based on subject or interest.
- Group admins can accept or reject join requests and manage group permissions.
- Scope includes options for public and private groups with invite-only access.

Real-Time Text Messaging

- Enable 1-on-1 chats and group chats using Socket.io for real-time updates.
- Include message delivery status (sent, delivered, seen).
- Implement message history, emojis, and file attachment support.

Voice and Video Communication

- Integrate WebRTC or similar technology for direct peer-to-peer voice and video calling.
- Users can start voice/video calls within groups or with individual study partners.
- Allow multi-user video conferences with mute/unmute and leave options.

Resource and File Sharing

- Users can upload and share academic materials such as PDFs, notes, or links within groups or chats.
- Include preview functionality for common file types.
- Add storage management and security measures (e.g., file type restrictions).

Post Sharing and News Feed

- Allow users to share study-related updates, announcements, and achievements through a common feed.
- Include features like commenting, liking, and sharing posts.

Invite and Friend System

- Users can send friend requests or share invite links to add peers to their study network or groups.
- Allow users to search for friends or students based on filters like subject, school, or interest.

Notifications and Reminders

- Real-time notification system for message alerts, group invites, file uploads, and reminders.
- Include push notifications for upcoming study sessions or pending group approvals.

Admin Panel

- For platform moderation and maintenance.
- Admins can view usage statistics, manage flagged content, and handle user complaints.
- Tools for banning or warning users for inappropriate behavior.

Scalability and Deployment

- Design the system with scalability in mind (e.g., adding more servers, cloud deployment).
- Host frontend on Vercel or Netlify, backend on Render or Heroku, and database on MongoDB Atlas.
- Ensure mobile responsiveness and accessibility compliance.

Security and Privacy

- Use HTTPS, input validation, and secure authentication to protect user data.
- Respect user privacy by limiting data access and providing privacy settings.

Out of Scope (for MVP)

- AI-based recommendations for study buddies (can be added in future versions).
- Gamification features like leaderboards or badges.
- Mobile app (current scope is limited to web platform).

3.2 Project Modules

The Virtual Study Buddy Platform is composed of multiple integrated modules, each handling a distinct functionality essential for facilitating effective online study collaboration. The modular structure ensures maintainability, scalability, and ease of future enhancements.

I. User Authentication Module

Functionality: Handles user registration, login, logout, and password recovery.

Features:

- Secure signup/login with email verification.
- Password encryption using hashing algorithms.
- Role-based access (e.g., admin, regular user).

II. User Profile and Dashboard Module

Functionality: Allows users to manage personal information and view their dashboard.

Features:

- Update name, profile picture, interests, and academic level.
- Dashboard displays recent messages, active groups, and upcoming sessions.

III. Study Group Module

Functionality: Enables users to create, manage, and join study groups.

Features:

- Create public or private groups.
- Send join/invite requests.
- Assign group admin roles.

IV. Real-Time Chat Module

Functionality: Provides instant messaging within groups and between users.

Features:

- One-to-one and group text chat.
- Real-time message delivery using WebSockets.
- Emoji and file sharing support.

V. Voice & Video Call Module

Functionality: Facilitates real-time audio and video communication.

Features:

- Peer-to-peer voice/video calling using WebRTC.
- Multi-user group calls.
- Call notifications and mute/unmute options.

VI. Post Sharing / News Feed Module

Functionality: Enables sharing of updates, thoughts, study tips, and announcements.

Features:

- Create text/image posts.
- Comment and like functionality.
- Display posts in reverse chronological order.

VII. Invite and Friend System Module

Functionality: Helps users expand their study network.

Features:

- Send/accept friend requests.
- Share group invite links.
- Search for users by name, interests, or academic level.

VIII. Notification Module

Functionality: Notifies users about important events and actions.

Features:

- Real-time alerts for new messages, invites, and posts.
- Optional push notification integration.

IX. File and Resource Sharing Module

Functionality: Facilitates the sharing of educational resources.

Features:

- Upload and download notes, links, and PDFs.
- Attach files in chat or posts.
- File size validation and preview options.

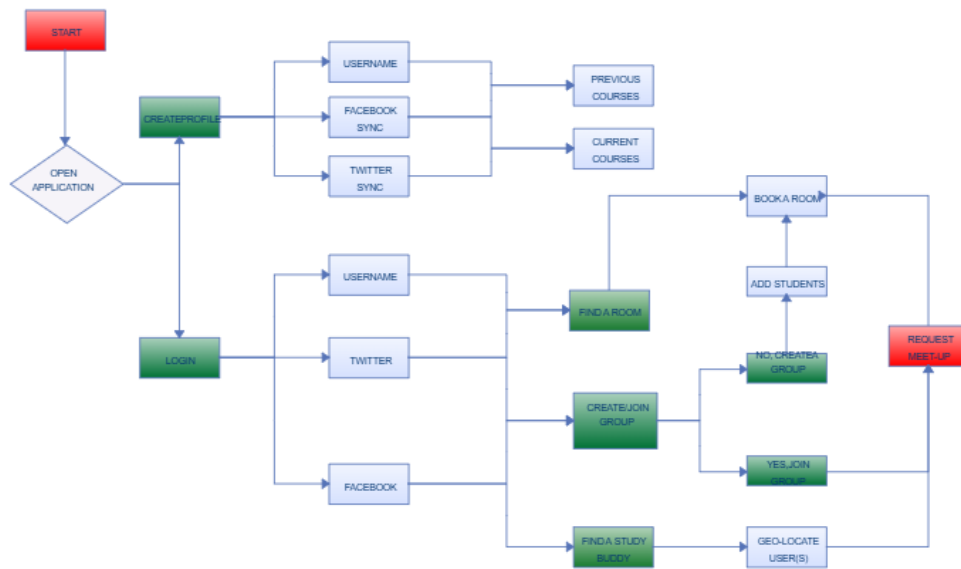
X. Admin Module

Functionality: Provides administrative controls to manage the platform.

Features:

- Monitor user activity and reported content.
- Block or warn users violating guidelines.
- View platform statistics and logs.

3.3 Context Diagram (High Level)



The flowchart represents the user interaction and functional workflow of a collaborative study application. The application is designed to allow users to create profiles, find or join study groups, book rooms for study sessions, and request meetups with peers.

1. Start

- The system flow begins with the user initiating the application.
- This represents the entry point into the application.

2. Open Application

Upon starting, the user is presented with two options:

- Create a new profile (for new users).
- Login (for returning users).

3. Create Profile

If the user selects Create Profile, the following actions are performed:

- The user enters a username to uniquely identify themselves in the system.
- Optionally, the user can sync their Facebook and Twitter accounts. This:
 - Enables easier login in the future.
 - May allow for importing friend lists or connecting with classmates.

The user is also prompted to provide academic data:

- Previous Courses – subjects the user has already studied.
- Current Courses – subjects the user is currently enrolled in.

4. Login

Existing users can log in via:

- Their Username (registered during profile creation).
- Or by using their Facebook or Twitter accounts.

This provides flexible access to the application for users with different preferences.

5. Post Login/Profile Functionalities

After logging in or registering, users can access core features of the system:

i. Find a Room

- Users can look for available study rooms based on availability or location.
- This helps in organizing or joining study sessions in physical spaces.

ii. Create or Join Group

- Users are prompted with a choice:
 - NO: If no group is joined, the user can create a new group.
 - The system then allows the user to add other students to this group.
 - YES: If the user chooses to join a group, they can browse or search for existing groups and join them directly.

iii. Find a Study Buddy

- Users can search for a study partner based on:
 - Similar courses
 - Study preferences
 - Location
- Helps in forming one-on-one collaborations.

6. Geolocation of Users

- After joining or creating a group, the user has the option to geolocate group members.
- This uses location data to show where other users are physically located.
- Helps in finding nearby users for offline meetups or discussions.

7. Book a Room

- If a group is formed and members are added, the system enables room booking.
- Users can schedule a time and book a space for group study sessions.

8. Request Meetup

- Once a room is booked or a group is formed, users can request a meetup with others.
- This may trigger notifications or confirmations to other users.
- Note: This represents the final collaborative action in the flow.

The flowchart illustrates a logical, step-by-step interaction for users in a study collaboration app. It covers user registration, social syncing, academic data entry, group formation, study room booking, geolocation, and meetup coordination. Each step is aimed at enhancing user engagement and enabling efficient academic collaboration.

4 Implementation Methodology

The implementation of the Virtual Study Buddy Platform follows a structured and iterative development methodology, ensuring systematic progress, effective collaboration, and high-quality results. The chosen approach combines elements of the Agile Software Development Life Cycle (SDLC) with modular component development.

I. Requirement Gathering and Analysis

Activities:

- Identifying the needs of students for online collaboration.
- Researching similar platforms for benchmarking.
- Defining clear functional and non-functional requirements.

Outcome: Software Requirements Specification (SRS) document.

II. System Design

Activities:

- Creating high-level and low-level architecture of the platform.
- Designing database schemas, system flow, and module interactions.
- UI/UX wireframing using Figma or Adobe XD for user interface layout.

Outcome: System Design Document and architecture diagrams.

III. Technology Stack Finalization

- **Frontend:** HTML, CSS, JavaScript, React.js, Tailwind CSS, Typescript
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Real-Time Communication:** Socket.io (for chat), WebRTC (for video/audio calls)
- **Authentication & Security:** JWT (JSON Web Tokens), Bcrypt.js, Clerk API

- **Version Control:** Git, GitHub

IV. Modular Development (Agile Sprint-based)

Each module was developed and tested in iterations or sprints, ensuring continuous delivery and integration.

Sprint 1 – User Authentication and Profile

- Implemented user login, signup, password encryption. Secure user authentication was built using JWT (JSON Web Tokens) for session handling, and bcrypt was used to hash passwords before storing them in the database, ensuring data protection.
- Created user profile dashboard with editable info. Users could view and update their profile information including name, bio, academic interests, and display picture. This provided a personalized experience and served as a foundation for matching with peers.

Sprint 2 – Study Groups & Friend System

- Enabled users to create/join groups and send friend invites. Users could initiate or join study groups, manage group membership, and build a friend list by sending and accepting invitations — fostering a sense of community.
- Integrated MySQL for storing user and group data. A relational database schema was designed in MySQL to manage users, groups, friendships, and permissions using foreign keys and normalization principles for data integrity and efficient querying.

Sprint 3 – Real-Time Messaging

- Used Socket.io for live chat functionality. Implemented WebSocket-based real-time messaging using Socket.io, allowing users to communicate instantly within group channels or private chats.
- Enabled one-on-one and group chat features. Developed scalable chat modules that supported both private messaging and group discussions, with chat histories stored in MySQL for persistence and retrieval.

Sprint 4 – Audio & Video Communication

- Integrated WebRTC and Peer.js for real-time communication. Leveraged WebRTC and Peer.js to enable low-latency, peer-to-peer video and audio calls between users for real-time virtual collaboration.
- Built UI controls for audio & video communication (mute, end, etc.). Designed an intuitive call interface with essential controls like mute/unmute, end call, and video toggle to enhance user experience during virtual study sessions.

Sprint 5 – Post Sharing and Resource Upload

- Develop Integrated WebRTC and Peer.js for real-time communication. Leveraged WebRTC and Peer.js to enable low-latency, peer-to-peer video and audio calls between users for real-time virtual collaboration.
- Built UI controls for audio & video communication (mute, end, etc.). Designed an intuitive call interface with essential controls like mute/unmute, end call, and video toggle to enhance user experience during virtual study sessions.

Sprint 6 – Notifications and Admin Controls

- Added notification system using Socket.io
Real-time push notifications alerted users about new messages, post interactions, friend requests, and group activity, keeping them engaged and informed.

V. Integration and Testing

To ensure the Virtual Study Buddy Platform was robust, user-friendly, and production-ready, a multi-phase testing strategy was followed throughout the development cycle. These phases helped verify the platform's functionality, compatibility, and performance under various scenarios, contributing to a stable and scalable release.

- **Unit Testing**

Each individual module and function—such as user registration, login authentication, message sending, file upload, and notification triggers—was tested independently. Automated and manual test cases were written to check for logic errors, validation issues, and expected output. This phase focused on isolating and resolving issues early in the development process to minimize cascading bugs.

- **Integration Testing**

Once individual modules passed unit tests, they were tested together to ensure smooth data flow and functional interconnectivity. Scenarios like syncing user profiles with group memberships, integrating the chat system with the notification engine, and aligning the file upload system with the post feed were thoroughly validated. Special attention was given to backend-frontend communication using APIs, database consistency, and user state management.

- **System Testing**

The complete system was tested as a whole to evaluate its compliance with the defined requirements. End-to-end tests ensured that features worked

together as intended in a live-like environment, covering real-time messaging, media streaming, and user interactions across different roles (admin, student, guest).

- **User Acceptance System (UAT)**

Real users were invited to interact with the platform under typical usage conditions. Their feedback on usability, interface design, speed, and navigation was used to fine-tune the user experience. Test users simulated actions such as forming study groups, hosting video calls, and sharing study materials, which provided valuable insights into practical usability and feature completeness.

- **Cross-Browser & Cross-Device Testing**

To ensure platform compatibility, the system was tested across multiple browsers (Chrome, Firefox, Edge, Safari) and devices (desktop, tablets, smartphones).

- **Performance & Load Testing:**

Simulated concurrent users and group activities were tested to evaluate system performance under load. Real-time chat and video calling features were assessed for latency, bandwidth usage, and system responsiveness to ensure the platform could handle multiple simultaneous users without lag or crashes.

- **Security Testing:**

Common vulnerabilities such as SQL injection, XSS, and unauthorized access were checked. Emphasis was placed on securing user data through encrypted communication, proper session handling, role-based access control, and file upload restrictions.

I. **Deployment and Maintenance**

A well-structured deployment and maintenance strategy was followed to ensure that the Virtual Study Buddy Platform is always available, secure, and responsive to user needs. This phase involved preparing the platform for real-world usage and supporting it post-launch with ongoing improvements.

I. **Deployment**

- **Frontend Hosting on Vercel/Netlify:**

The frontend of the platform, built using modern JavaScript frameworks (like React), was deployed on Vercel or Netlify for high performance, global CDN

support, and seamless CI/CD workflows. This allowed rapid deployment from the development branch and zero-downtime updates.

- **Backend API and WebSocket Server Hosting on Render/Heroku:**

Backend services, including RESTful APIs and real-time WebSocket communication using Socket.io, were deployed on platforms like Render or Heroku. These platforms provided scalability, managed environments, and simplified infrastructure management.

- **MySQL Database Hosting:**

The relational database was hosted either on a cloud MySQL provider (e.g., PlanetScale, Railway, or ClearDB on Heroku) or on a VPS with remote access. It ensured structured data storage for users, study groups, messages, and other entities with regular backups and encrypted connections.

II. Post-Deployment Maintenance

- **Bug Fixes and Security Patching:**

Continuous monitoring was implemented to detect and resolve bugs promptly. Routine updates were released to patch any discovered vulnerabilities, ensuring the platform remained secure against threats like SQL injection, XSS, or unauthorized access.

- **User Feedback Collection:**

Feedback mechanisms were integrated within the platform to allow users to report issues or suggest improvements. Collected insights were used to prioritize new features, refine the UI/UX, and adapt the system based on real user behavior.

- **Performance Monitoring and Logs:**

Tools such as Log Rocket, Sentry, or server-side logging systems were used to monitor server health, catch frontend/backend errors, and maintain uptime. This ensured optimal performance and quick responses to any service disruption.

- **Scalability Planning:**

Prepared the platform to handle future growth by using auto-scaling services, optimizing SQL queries, indexing the database, and planning horizontal scaling of backend services.

II. Documentation

Comprehensive documentation was developed to support all stakeholders involved with the Virtual Study Buddy Platform — including developers, users, evaluators, and future maintainers. This ensured the platform was understandable, maintainable, and scalable beyond its initial release.

- **Technical Documentation for Code and Database:**

Detailed technical documentation was prepared covering the overall system architecture, technology stack, API endpoints, database schema (MySQL), authentication flow, and real-time communication setup. Inline code comments, README files, and setup guides were also included to assist future developers in onboarding and contributing effectively.

- **User Manual for End-Users:**

A step-by-step user guide was created to help users navigate the platform's features — such as account creation, joining and creating study groups, sending and receiving messages, initiating and participating in audio/video calls, sharing academic resources like notes and documents, and managing their personal profiles, including updating academic information and syncing social media accounts. The manual is designed to ensure a user-friendly experience by providing clear instructions and illustrations for each feature.

- **Report and Presentation Slides for Academic Submission:**

A well-organized final report and presentation slides were compiled, summarizing the project's objective, problem statement, methodology (Agile sprints), system design, development process, testing strategies, and outcomes. These materials were tailored for academic evaluation and oral defense.

- **User Feedback Integration and Future Roadmap:**

Feedback collected from real users during UAT and post-deployment was documented and categorized. Based on insights, a roadmap was drafted which included:

- i. Enhancing accessibility and UX based on student suggestions
- ii. Adding AI-driven recommendations for study groups, resources, and topic suggestions
- iii. Planning mobile app development for Android and iOS to improve accessibility and engagement

5 Technologies to be used

5.1 Software Platform

a) Front-end

▪ React.js

The platform's user interface is built using React.js, a powerful JavaScript library for building responsive and component-based front-end applications. React allows for dynamic updates, real-time data rendering, and seamless integration with APIs and WebSocket communication.

b) Back-end

▪ Node.js with Express.js

The server-side logic, API handling, and WebSocket communication are implemented using Node.js along with the Express.js framework. This stack provides scalability, non-blocking I/O operations, and robust middleware support—ideal for real-time applications like chat and video calling.

5.2 Hardware Platform

To develop, test, and run the platform efficiently, the following minimum hardware and system configuration is recommended:

- **RAM:** 4 GB or higher
- **Storage:** 10-15 GB
- **Operating Systems Supported:** Windows, Linux, macOS
- **Code Editor:** Visual Studio Code (VS Code)
- **Browser:** Google Chrome (for development and testing)

5.3 Development Tools

The development and deployment of the Virtual Study Buddy Platform involved the use of the following tools across various stages:

▪ Visual Studio Code (VS Code):

Primary code editor used for front-end and back-end development, with extensions for linting, Git integration, and debugging.

▪ Postman:

For testing API endpoints and verifying backend responses during development.

- **Git & GitHub:**
Version control system and repository hosting for collaborative development and source code management.
- **Render / Vercel / Netlify:**
Deployment platforms used for hosting the backend (Render/Heroku) and frontend (Vercel/Netlify).
- **MySQL Workbench or phpMyAdmin:**
GUI tools used for designing and managing the MySQL database schema and queries.

6 Advantages of this Project

The In the digital age where remote education and online collaboration have become mainstream, the Virtual Study Buddy Platform stands out as a transformative solution. It bridges the gap between traditional educational tools and modern student needs by offering real-time, interactive, and flexible ways to collaborate. Below is an in-depth look at the platform's key advantages:

I. Enhanced Collaboration

One of the primary goals of this platform is to foster teamwork among students beyond classroom boundaries. By allowing users to connect from anywhere and form study groups around common interests or courses:

- Students can collaborate on assignments, prepare for exams, or discuss lecture topics.
- The inclusion of voice and video chat makes discussions more engaging and mimics in-person group study sessions.
- Students who might feel isolated during remote learning can now feel connected, boosting morale and motivation.

II. Time and Cost Efficiency

Traditional in-person group studies or tutoring sessions often require travel, coordination, and extra resources. This platform eliminates those burdens by:

- Offering a virtual space that's always accessible, reducing commute times and scheduling issues.
- Being either free or extremely low-cost, it levels the playing field for students who may not have access to expensive educational tools or tutors.
- Providing built-in collaboration tools that reduce the need for multiple third-party apps.

III. Personalized Learning

Every student learns differently, and this platform supports that by enabling a customizable learning experience:

- Users can join or create study circles based on specific subjects, difficulty levels, or exam preparations.
- They can also choose who to collaborate with, building a comfortable and focused study environment.
- This flexibility leads to better engagement, as students feel more ownership over their learning process.

IV. Real-Time Communication

Modern learning often requires instant feedback and immediate clarification of doubts. This platform provides:

- Instant messaging to quickly discuss questions or share quick tips.
- Audio and video calling for more complex explanations, group meetings, or peer tutoring sessions.
- Real-time communication tools simulate a classroom environment, keeping users motivated and engaged.

V. Resource Sharing

A critical part of academic success is access to high-quality resources. The platform enables users to:

- Upload and download notes, documents, images, PDFs, or video tutorials
- Share useful links to educational websites, tools, or articles.
- Foster a culture of peer learning, where students help each other grow by sharing their strengths and materials.

VI. Easy Group Management

Managing online groups can be difficult without proper tools. This platform solves that by offering:

- Features to create study groups, invite members, assign roles (like admin or moderator), and manage discussions effectively.
- Group creators (students, teachers, or mentors) can control who joins, what gets posted, and how interactions take place.

- This structure ensures discussions remain productive, respectful, and focused.

VII. Accessible Anytime, Anywhere

Flexibility is crucial in modern education. This platform is:

- Web-based, meaning no need to install heavy software — simply log in from a browser.
- Compatible with multiple operating systems and devices (Windows, Linux, macOS, mobile phones).
- Perfect for students on the move, such as those juggling part-time jobs or living in remote areas.

VIII. User-Friendly Interface

Tech-savviness shouldn't be a barrier to collaboration. Thanks to React.js, the platform features:

- A clean and intuitive UI that guides users naturally through features.
- Responsive design that adapts to screens of all sizes, from smartphones to desktops.
- Clear icons, smooth transitions, and minimal clicks make for a hassle-free user experience, even for first-time users.

IX. Scalable Architecture

The system is designed to grow as its user base expands:

- Technologies like Node.js and MySQL ensure stable performance even with high traffic.
- Modular design allows developers to add features like quizzes, file storage, or mobile apps without reworking the entire platform.
- This scalability makes it ideal for deployment in schools, colleges, and global student communities.

X. Secure and Private

Security is vital, especially for young users. The platform ensures:

- Encrypted user data and secure login processes, protecting against unauthorized access.

- Private communication channels, giving users peace of mind while sharing personal or academic information.
- Optional admin/moderator roles to monitor and moderate activity, creating a safe and respectful space for everyone.

The Virtual Study Buddy Platform not only facilitates academic collaboration — it reimagines how students connect, communicate, and learn in the digital age. By integrating smart design with educational purpose, it empowers learners, encourages sharing, and builds communities of support and success.

7 Assumptions

In the development and deployment of the Virtual Study Buddy Platform, the following assumptions have been made to ensure project feasibility, technical planning, and appropriate expectation setting:

1. Internet Connectivity

It is assumed that all users (students and admins) will have a stable internet connection to access the platform's real-time features such as messaging, video calling, and media sharing.

2. Device Accessibility

Users will have access to a device (laptop, desktop, tablet, or smartphone) with a modern web browser that supports React.js applications and WebRTC for real-time communication.

3. Technical Familiarity

Users will possess basic digital literacy skills (e.g., signing up, navigating a web app, joining groups), as the platform is designed to be intuitive but not heavily tutorial-dependent.

4. Privacy and Safety

Users are expected to follow community guidelines, and a basic level of trust is assumed among peers. However, moderation tools and admin features are built in to manage misuse or inappropriate behavior.

5. Scalability Requirements

Initial deployment is assumed to serve a limited number of users (e.g., a small academic institution or beta test group). Scaling strategies will be implemented as demand grows.

6. Cross-Platform Support

The platform is expected to function across all major operating systems (Windows, Linux, macOS) and browsers (Chrome, Firefox, Edge, Safari) without the need for platform-specific configurations.

7. User Verification

Email-based user authentication is assumed to be sufficient for verifying user identities, especially during early deployment phases.

8. Data Volume

It is assumed that the initial volume of messages, media uploads, and user-generated content will be manageable within the provided backend (MySQL and file storage), and growth will be monitored.

9. Browser Permissions

Users will allow access to microphone and camera when needed for video/audio communication, which is critical for WebRTC functionality.

10. No Third-Party Login Integration (Initially)

It is assumed that users will register and log in using the platform's custom authentication system, with optional third-party OAuth integration (Google/Facebook) planned for later phases.

11. Server Uptime and Hosting Stability

It is assumed that the hosting services (e.g., Vercel, Heroku, or any cloud provider) will provide consistent uptime and minimal outages to ensure availability of the platform.

12. Maintenance Support

It is assumed that a small development or IT team will be available post-deployment to handle minor bug fixes, updates, and server monitoring.

13. Legal and Compliance Considerations

It is assumed that the platform will comply with basic privacy policies (e.g., GDPR, if necessary), and that users agree to terms of service and community guidelines.

14. User Base Growth is Gradual

Initial adoption of the platform is expected to be moderate, allowing time to scale infrastructure, optimize performance, and integrate feedback before a larger rollout.

15. Limited Customization by End-Users

It is assumed that users will use the platform as designed without extensive need for interface customization (themes, branding, plugins), unless introduced in future versions.

16. Browser Compatibility

It is assumed that users will access the platform through modern browsers that support ECMAScript 6+, WebRTC, and other essential web technologies.

17. English as Primary Language

It is assumed that English will be the default language for interface and communication. Multilingual support may be added based on user demand in future versions.

8 Future Scope and further enhancement of the Project

The Virtual Study Buddy Platform demonstrates immense potential for growth, especially with the increasing reliance on virtual learning and collaborative environments in education. The following enhancements and features outline the roadmap for scaling, improving engagement, and broadening the platform's impact:

I. Integration with Learning Management Systems (LMS)

- Seamless integration with LMS platforms such as Moodle, Canvas, or Google Classroom would allow students and educators to manage courses, assignments, and grades directly within the study groups.
- This would consolidate learning tools and streamline user workflows.

II. AI-Based Study Matchmaking

- Incorporating machine learning algorithms to intelligently recommend study partners or groups based on factors like academic interests, learning styles, past interactions, and schedules.
- Personalized suggestions would foster more productive and compatible collaborations.

III. Calendar and Task Scheduling

- Introduction of a shared calendar system with task reminders, event scheduling, and to-do lists for both individuals and groups.
- Enables users to plan sessions, track deadlines, and set group goals efficiently.

IV. Gamification and Rewards

- Implementing gamification elements such as badges, points, rankings, and streak counters to incentivize consistent participation.
- Promotes engagement by recognizing active learners and collaborative contributors.

V. Analytics Dashboard

- Development of analytics tools for both students and educators to gain insights into study patterns, participation levels, and time spent in discussions.

VI. Integration with Cloud Storage Services

- Adding support for services like Google Drive, Dropbox, and OneDrive for easy file sharing and cloud storage.
- Users can directly access and manage study materials without leaving the platform.

VII. Mobile Application Development

- Building native or hybrid mobile apps (using React Native or Flutter) to offer a seamless mobile experience.
- Increases platform reach and accessibility for users on smartphones and tablets.

VIII. AI-Powered Doubt Resolution Bot

- Deploying an AI chatbot capable of answering common queries, recommending study resources, and directing users to relevant channels or groups.
- Acts as a 24/7 support system for learners.

IX. Multi-Language Support

- Introducing support for multiple languages to cater to users from various regions and backgrounds.
- Enhances inclusivity and allows better communication among non-English-speaking students.

X. Enhanced Security and Privacy Features

- Incorporating end-to-end encryption for messaging and video calls to protect user privacy.
- Adding parental controls, moderation tools, and reporting features to ensure a safe environment, especially for underage users.

XI. Expansion to Institutional Use

Customizing the platform for deployment in schools, colleges, and coaching centers, offering features such as:

- Institution-specific group creation.
- Educator dashboards.
- Class attendance tracking.
- Group grading and announcements.

9 Project Repository Location

S#	Project Artifacts (softcopy)	Location (Mention Lab-ID, Server ID, Folder Name etc.)	Verified by Project Guide	Verified by Lab In-Charge
1.	Project Synopsis Report (Final Version)		Name and Signature	Name and Signature
2.	Project Progress updates		Name and Signature	Name and Signature
3.	Project Requirement specifications		Name and Signature	Name and Signature
4.	Project Report (Final Version)		Name and Signature	Name and Signature
5.	Test Repository		Name and Signature	Name and Signature
6.	Project Source Code (final version) with executable		Name and Signature	Name and Signature
7.	Any other document		Name and Signature	Name and Signature

10 Definitions, Acronyms, and Abbreviations

Abbreviation	Description
API	Application Programming Interface - A way for different software components to communicate.
Auth	Authentication - Process of verifying a user's identity.

11 Conclusion

The Virtual Study Buddy Platform represents a significant advancement in the landscape of modern education, aiming to redefine how students collaborate, communicate, and support each other in their academic pursuits. By utilizing digital technologies to bridge the gap between students seeking academic companionship, the platform creates an ecosystem where learning becomes more personalized, engaging, and socially enriched.

At its core, the platform leverages intelligent algorithms to match users based on shared courses, learning styles, academic goals, and availability. This personalized approach ensures that users are paired with compatible study partners or groups, thereby enhancing the quality and relevance of their collaborative experiences. The integration of features such as real-time messaging, audio and video calling, shared resource libraries, and collaborative tools further empowers students to interact dynamically and effectively in both synchronous and asynchronous learning environments.

One of the platform's most impactful features is its goal-setting module, which allows students to define, track, and manage academic objectives with their study partners. This fosters not only a sense of structure and discipline but also accountability and mutual motivation—key factors in sustained academic performance. Additionally, the resource sharing capabilities support the exchange of notes, study materials, and important academic content, promoting collective learning and knowledge building.

The platform also emphasizes community engagement, encouraging students to form study circles, participate in discussion forums, and organize virtual or physical meetups. These interactions contribute to the development of soft skills, increase exposure to diverse perspectives, and foster a sense of belonging within an academic community. Through these socially interactive components, the platform goes beyond the role of a mere utility tool and evolves into a comprehensive support system for learners.

From an educational impact standpoint, the implementation of the Virtual Study Buddy Platform signifies a shift toward student-centered learning—empowering users to take charge of their academic journeys. It aligns well with the contemporary needs of hybrid and remote education, especially in a world where traditional classroom boundaries are increasingly fluid. By supporting collaborative learning outside the conventional school or university setup, the platform enhances accessibility, inclusivity, and flexibility in education.

Looking ahead, the project holds substantial promise for further development and scalability. Potential future enhancements include:

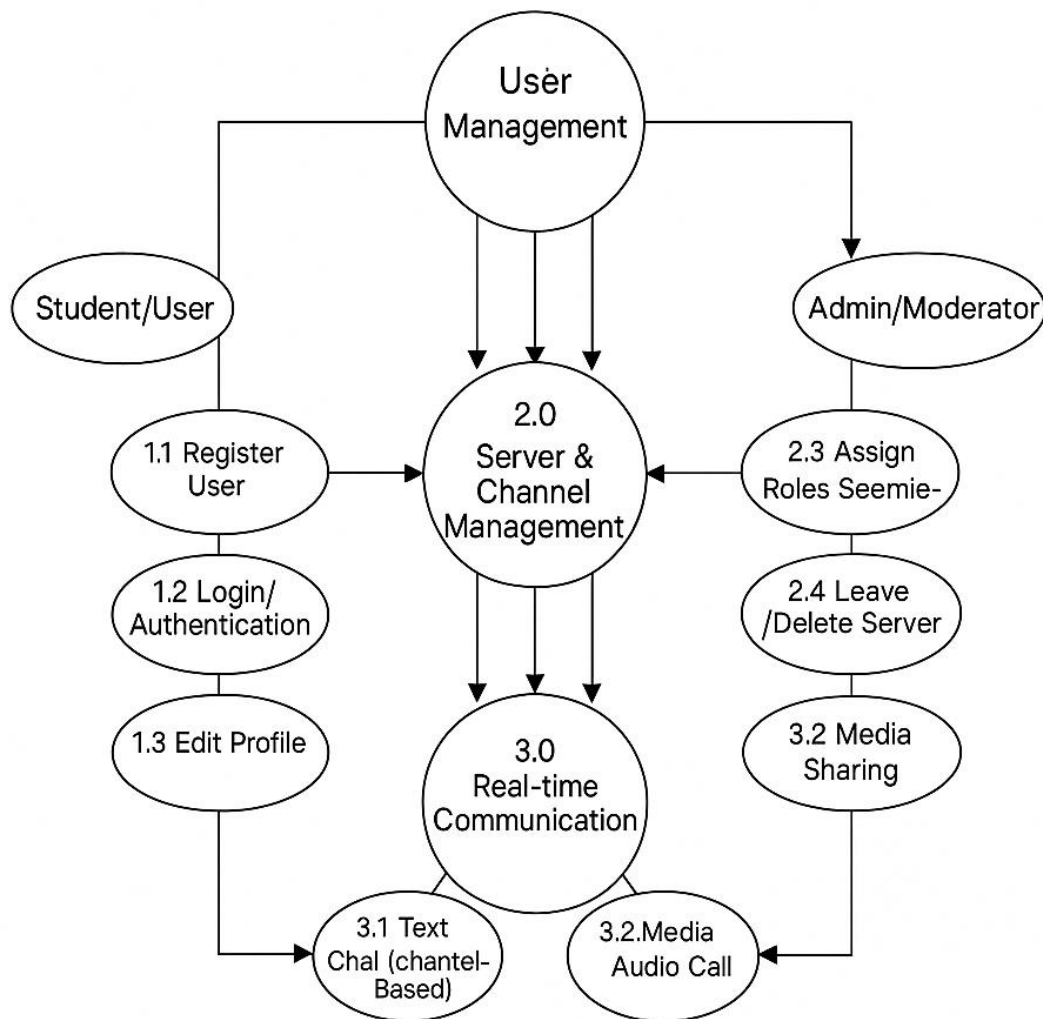
Development of a mobile application to provide ubiquitous access and seamless on-the-go interaction. Advanced communication features such as shared whiteboards, real-time annotation, and AI-powered transcription for study sessions. Integration with external educational platforms (e.g., Google Classroom, Moodle, Coursera), enabling automatic syncing of courses and schedules.

Gamification elements to boost engagement, such as achievement badges, study streaks, and collaborative challenges. Data analytics dashboards for performance tracking and personalized study recommendations. These future upgrades would not only improve user experience but also solidify the platform's relevance in evolving academic and technological contexts.

In conclusion, the Virtual Study Buddy Platform aspires to create a vibrant, dynamic, and inclusive educational community. By promoting knowledge exchange, peer support, and mutual growth, it aims to enhance both academic success and the development of lifelong learning habits. As the platform continues to evolve, it has the potential to make a lasting and transformative impact on how students learn, connect, and grow—paving the way for a more collaborative and empowered future in education.

Annexure A

Data Flow Diagram (DFD)



Level 3 DFD for PeerLynx

The Level 3 DFD for PeerLynx provides a detailed representation of the internal processes and data flow within the platform. PeerLynx is a virtual collaboration platform tailored for students to connect, communicate, and collaborate in real-time, inspired by modern tools like Discord.

This DFD decomposes the core functionalities into three major subsystems:

- **User Management**
- **Server & Channel Management**
- **Real-time Communication**

Each subsystem includes subprocesses that illustrate how users interact with the system and how data flows through various components.

1. User Management

This section of the DFD handles all user-related operations from the point of registration to profile management.

Subprocesses:

- **Register User:** Students create accounts by submitting personal information.
- **Login/Authentication:** Verifies user credentials to grant platform access.
- **Edit Profile:** Enables users to update their profile details such as avatar, bio, or subject interests.

Data Stores:

User Database: Stores all user credentials, profiles, and authentication tokens.

2. Server & Channel Management

This module allows users to organize their academic activities through dedicated study servers and topic-specific channels.

Subprocesses:

- **Create Study Server:** Users initiate a collaborative environment by creating a server.
- **Join Server:** Enables users to join existing study groups.
- **Assign Roles:** Admins assign roles such as moderator or member for better server control.
- **Leave/Delete Server:** Allows users to leave a server or admins to delete it entirely.

Data Stores:

Server & Channel Registry: Maintains information on servers, channels, and user roles.

3. Real-time Communication

This section covers synchronous collaboration within the platform.

Subprocesses:

- **Text Chat (Channel-Based):** Facilitates real-time messaging in topic-specific channels.
- **Media & Audio Call:** Supports voice/video calls and file/media sharing.

External Systems:

External APIs: Integrated services like WebRTC for real-time calls and cloud storage for media uploads.

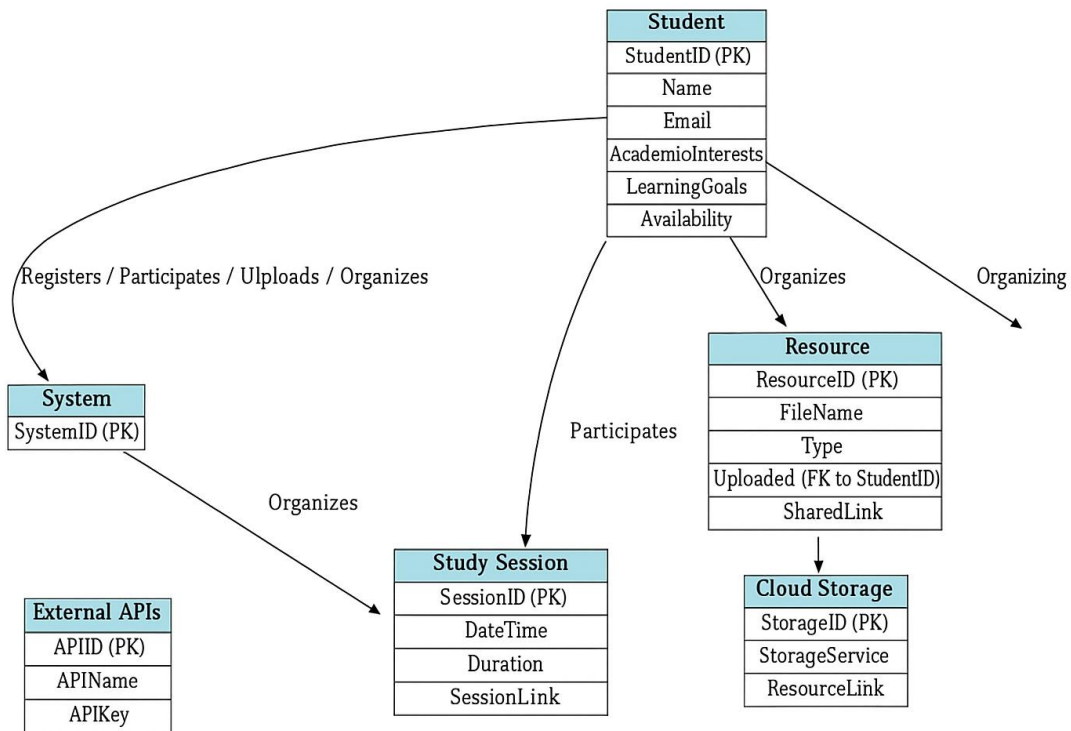
External Entities:

Student/User: The primary actor interacting with all system processes.

Admin/Moderator: Has special privileges for managing servers and assigning roles.

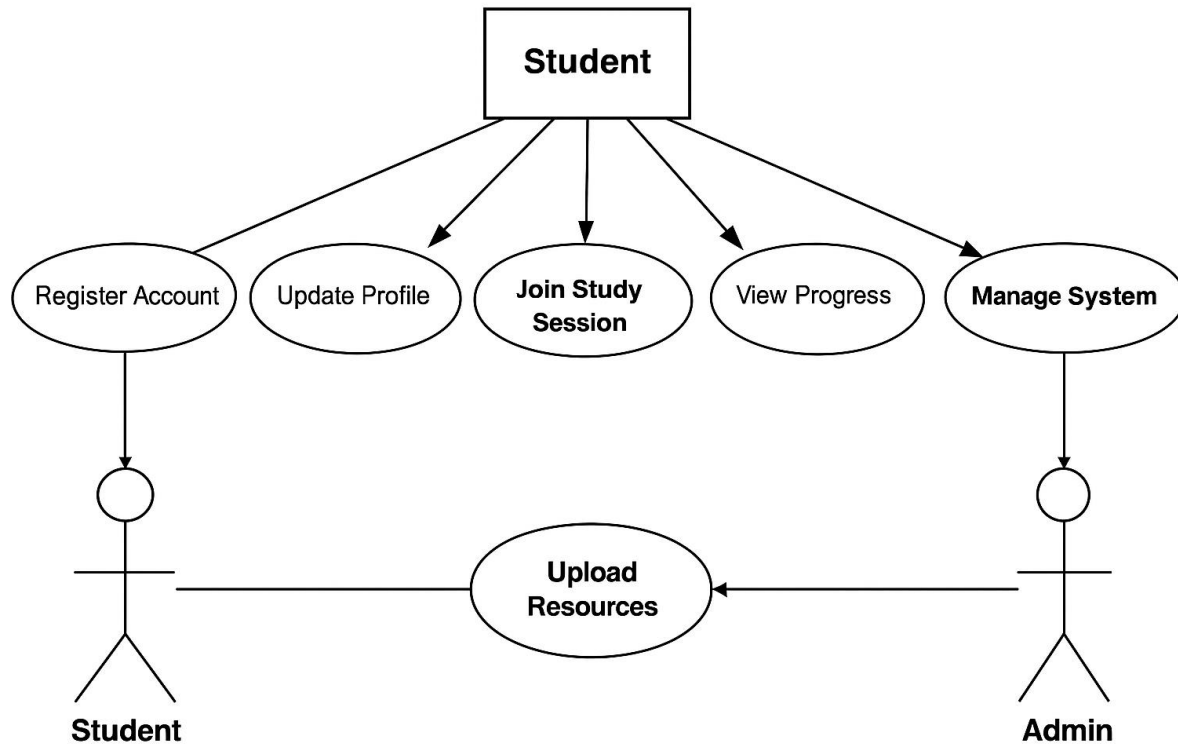
Annexure B

Entity-Relationship Diagram (ERD)



Annexure C

Use-Case Diagram (UCD)



Annexure D

Data Dictionary (DD)

(Mandatory)

Example:

User Table (USR)

Fields	Data type	Description
USR-Name	Text	Admin name
USR-Password	Text	Admin password
USR-Contact-No	Number	Admin Contact
USR-Address	Text	City

Supplier Table (SUPP)

Fields	Data type	Description
SUPP-ID	Number	Supplier ID
SUPP-Name	Text	Supplier Name
SUPP-Address	Text	Supplier Address
SUPP-Contact	Number	Supplier Contact
SUPP-Credit-Limit	Number	Credit Limit


Annexure E

Screen Shots

Login Page:



Sign in
to continue to PeerLynx

 Continue with Google

or

Email address

CONTINUE

No account? [Sign up](#)

Personalized Server Creation:



Customize your server

Give your server a personality with a name and an image. You can always change it later.

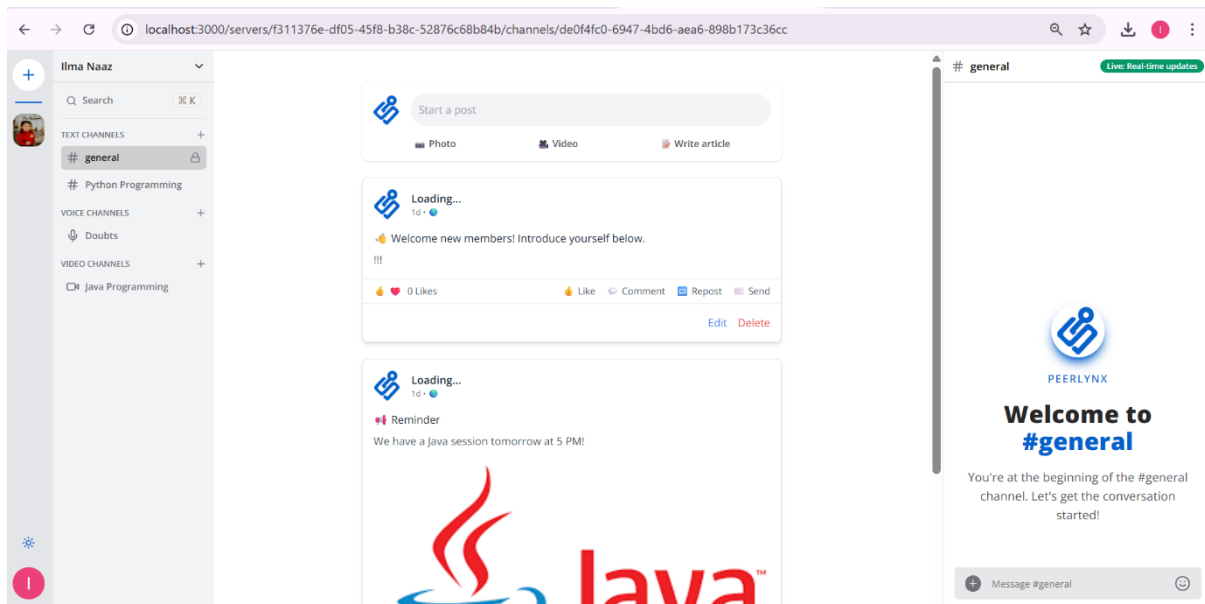
Choose files or drag and drop
image (JPG)

SERVER NAME

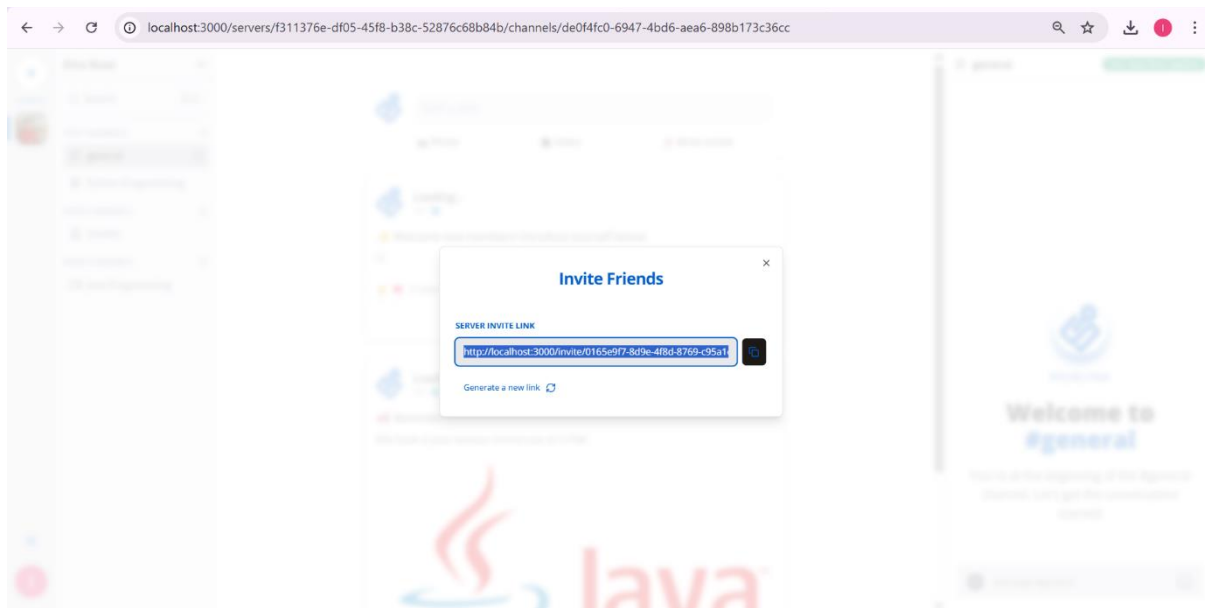
Enter server name

Create

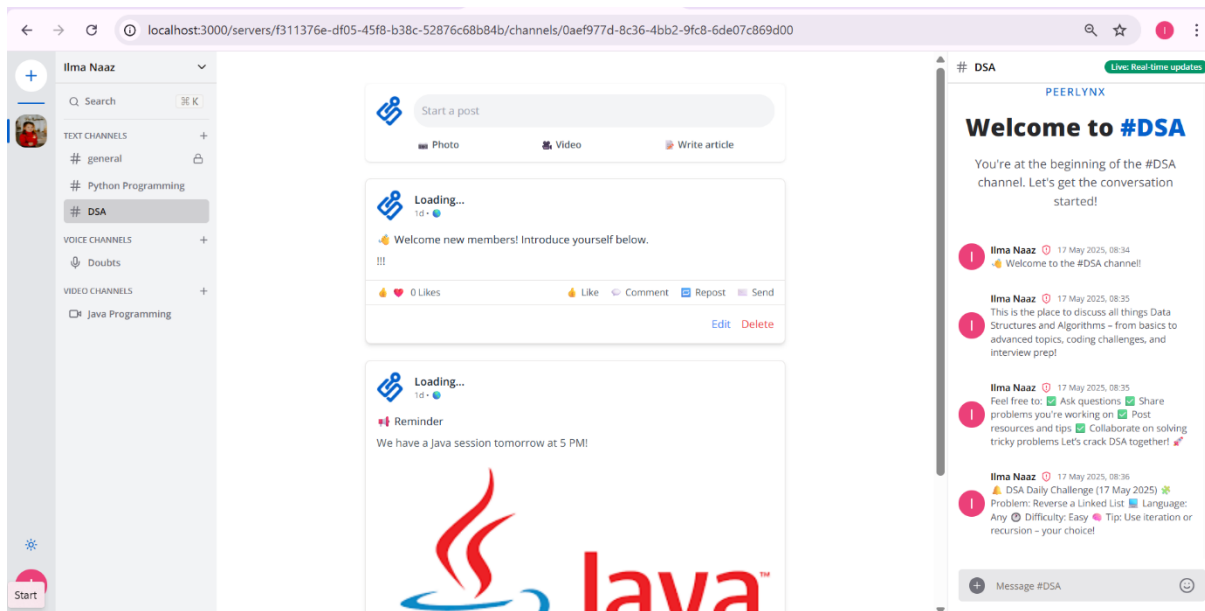
Home Page:

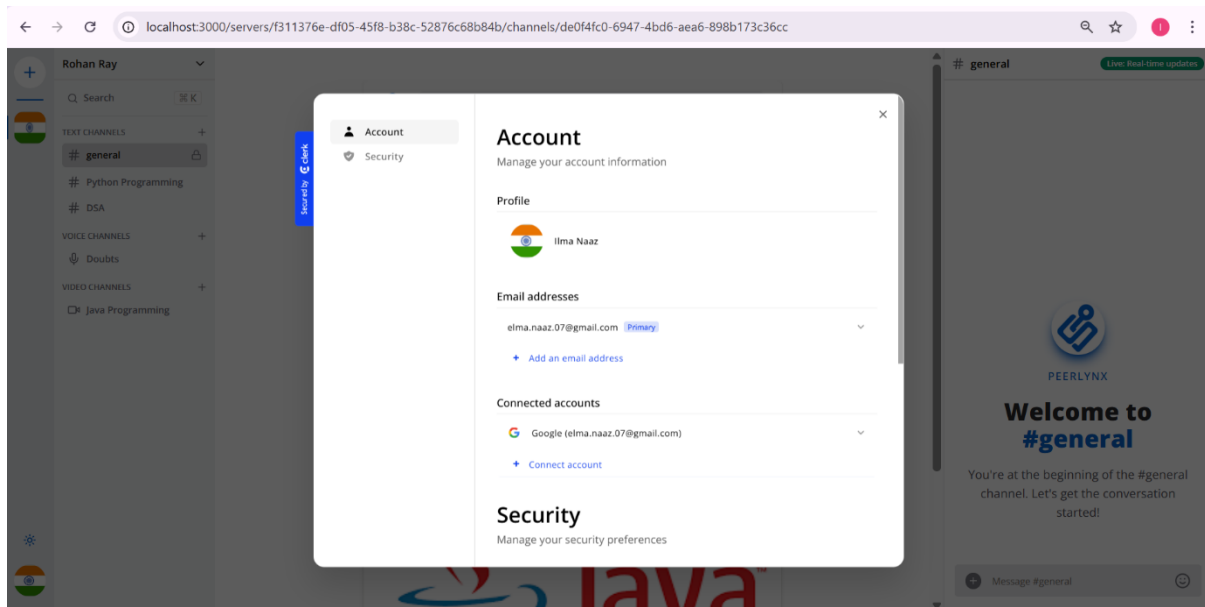


Invitation Page:



Text Channel:





Video Channel:

